

基于 Kinect 的手部关节位置推断方法的研究

郭小焕¹ 全太锋² 潘英杰¹ 马浩¹

¹(重庆邮电大学 重庆 400065)

²(信号与信息处理重庆市重点实验室 重庆 400065)

摘要 针对 Kinect 摄像头在追踪手部骨骼关节时识别精度低的问题,提出一种用于纠正该跟踪问题的手部关节位置推断方法。通过 Kinect 获取深度信息并进行预处理,获得人体区域的深度图像;通过阈值分割获得手臂部分的深度图像;基于正向运动学算法获得腕部关节的 3D 空间位置,提取出手部的深度图像后利用质心公式计算手部关节的空间坐标。实验结果证明,该方法可以在测试者挥动手臂时较为准确地推断出手部关节位置以及腕部关节和肘部关节的几何关系。

关键词 三维空间 阈值分割 深度信息 手部关节 位置推断

中图分类号 TP391.41

文献标志码 A

DOI:10.3969/j.issn.1000-386x.2020.10.026

POSITION INFERRING OF HAND JOINTS BASED ON KINECT

Guo Xiaohuan¹ Quan Taifeng² Pan Yingjie¹ Ma Hao¹

¹(Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

²(Key Laboratory of Signal and Information Processing of Chongqing, Chongqing 400065, China)

Abstract Aiming at the problem that the recognition accuracy of the Kinect camera is low when tracking the hand joints, this paper presents a position inferring method of hand joints for correcting the tracking problem. The depth information was obtained by Kinect and preprocessed to get the depth image of human body region; the depth image of the arm part was obtained by threshold segmentation; based on the forward kinematics algorithm, the 3D spatial position of wrist joint was obtained; after extracting the depth image of hand, the spatial coordinate of hand joint was calculated by the centroid formula. The experimental results show that our algorithm can more accurately infer the position of the hand joints and the geometric relationship between the wrist joints and the elbow joints when the tester swings the arms.

Keywords 3D space Threshold segmentation Depth information Hand joints Position inferring

0 引言

Kinect 是由微软开发的可以提供彩色和深度图像的 3D 体感摄影机。Kinect 能够以每秒 30 帧的频率实时检测和跟踪人体的 25 个关节并返回其 3D 空间坐标,实现无标记的人体关节空间位置信息的获取^[1]。人体姿态估计一直是计算机视觉的主要研究方向之一^[2],特别是对手部的捕捉,被广泛应用于行为识别、人机交互、医疗教育等领域。因此,许多学者在进行手

势识别和动作捕捉^[3]等研究中使用 Kinect 进行深度信息的采集。

Kinect 的关节捕捉不是绝对的精确。一些学者对 Kinect 的捕捉精度进行了研究。Xu 等^[4]在使用 Kinect 提取步态参数时,认为数据的偏差是由于 Kinect 在检测踝部关节时产生了错误的识别。Wang 等^[5]研究了 Kinect 在进行骨骼追踪中的关节位置精度,发现检测到的关节位置平均偏移量在 50 ~ 100 mm 之间。对于如何矫正捕捉精度的偏差,相关的研究较少。Osman 等^[6]提出使用正向运动学判断踝关节的位置。Srisen

等^[7]提出使用光流法进行手部关节的校正,但其需要密集的图像采集,对光照条件有较大的要求,会严重影响人体捕捉的实时性。Rosario 等^[8]使用粒子滤波器来矫正 Kinect 输出的关节旋转角度。李昕迪等^[9]提出人体单关节节点的修复算法,利用上下关节的位置关系推断被遮挡的中间关节的位置,该方法对手部关节的位置矫正效果较差。

针对 Kinect 对手部关节位置判断错误的问题,本文提出一种判断手部关节位置的新方法:通过图像分割提取手臂和手部的深度图像,利用正向运动学公式推断出肘部关节和腕部关节的位置后,再分割出手部的深度图像并计算出质心,解决手部关节位置无法正确判断的问题。

1 手部关节位置推断方法

本文针对 Kinect 摄像头在追踪手部骨骼关节时识别精度低的问题,提出一种用于纠正该跟踪问题的手部关节位置推断方法。首先,通过 Kinect 获取深度信息并进行前景提取;然后通过阈值分割获得手臂部分的深度图像;接着基于正向运动学算法判断腕部关节的 3D 空间位置,以此区分手臂与手部的深度图像;最后对提取出的手部深度图像利用质心公式推断手部关节的空间坐标。图 1 为手部关节位置判断方法的流程。

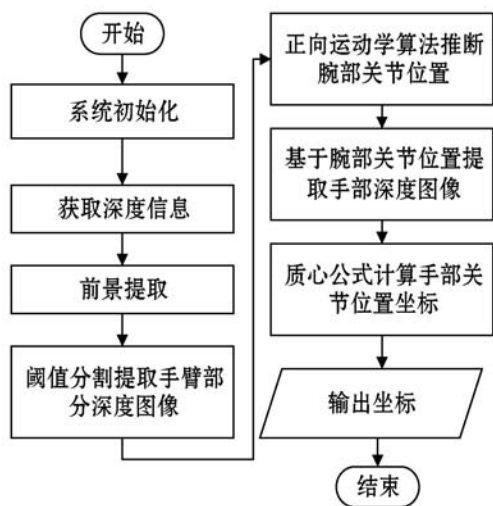


图 1 手部关节位置判断方法流程图

2 深度图像的获取与处理

2.1 Kinect 传感器

第二代 Kinect(以下简称 Kinect)是微软于 2014

年推出的一款 3D 体感设备^[10]。Kinect 硬件结构如图 2 所示。该传感器获得的深度信息,表示物体与传感器的实际距离。Kinect 可以进行实时人体骨骼追踪,追踪的关节如图 3 所示。

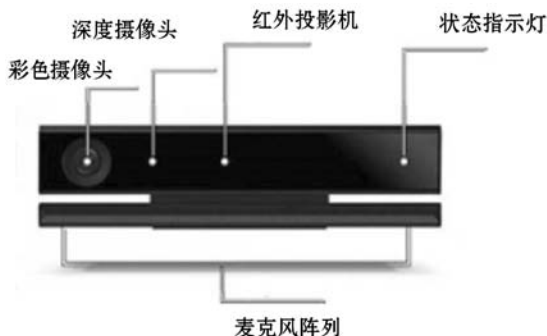


图 2 Kinect 传感器的硬件结构



图 3 骨骼追踪的关节示意图

2.2 人体深度图像获取

Kinect 提供了使用深度数据进行前景分割的功能^[11]。深度图像对比图如图 4 所示,图 4(a)是原始深度图像;通过 Kinect 为每个对象随机分配的索引号可以获得相应的掩码深度,从而提取出人体的深度图像,结果如图 4(b)所示;原始的深度图像目标边缘有很多噪点,如果直接处理粗糙图像,则边缘噪声对后续骨架的提取和关节判断具有很大的影响,因此本文使用深度中值滤波方法,在去除孤立噪声点的同时保留目标边缘信息^[12],平滑后的图像如图 4(c)所示。



(a) 原始深度图像 (b) 掩码深度图像 (c) 平滑后的深度图像

图 4 深度图像对比图

3 手臂深度图像的提取

人体的运动具有一定的遮挡性,在手臂挥动时, Kinect 无法从图 4(c) 的人体深度图像中正确提取出手部的正确位置。为了解决该问题,需要将手臂与手部的深度图像分离,以获得精确的手部深度图像。首先使用金字塔分割算法将深度图像划分为均匀的深度块^[13];然后设置深度信息直方图 H_d , 根据每个块的深度值来绘制深度图像直方图;再通过阈值分割来划分人体和手臂。通常人体的躯干部分深度值是深度信息统计中最大的,因此可以将删除的部分定义如下(假设有 S 个深度块):

$$D(q) = \max_{1 \leq p \leq S} \{D(H_d)\} \quad q \in \zeta \quad (1)$$

式中: ζ 表示除手臂之外的部分; q 表示深度图像中深度块的某一像素。通过选择阈值来划分身体部位(区域 A)和手臂部位(区域 B)。假设存在 L 个深度值,并且剩余深度块的总和是 N ,深度值为 i 的深度块数量是 n_i ,每个深度值出现的概率为 P ,得到以下公式:

$$N = \sum_{i=0}^{L-1} n_i \quad P_i = \frac{n_i}{N} \quad (2)$$

区域 A 和区域 B 的出现概率满足以下等式:

$$P_A = \sum_{i=t+1}^{L-1} P_i \quad P_B = \sum_{i=0}^t P_i = 1 - P_A \quad (3)$$

式中: t 是假设的选定阈值。区域 A 和区域 B 的组间方差满足以下等式:

$$\omega_A = \sum_{i=t+1}^{L-1} iP_i/P_A \quad \omega_B = \sum_{i=0}^t iP_i/P_B \quad (4)$$

$$\omega_0 = P_A \omega_A + P_B \omega_B = \sum_{i=0}^{L-1} iP \quad (5)$$

$$\sigma^2(t) = P_A (\omega_A - \omega_0)^2 + P_B (\omega_B - \omega_0)^2 \quad (6)$$

式中: ω_A 和 ω_B 表示区域 A 和区域 B 的平均深度值; ω_0 表示全部人体深度图像的平均深度值; $\sigma^2(t)$ 代表两个区域的方差,表示了深度分布均匀性的度量。手臂部分的提取是根据深度信息的特征来选择阈值,从而最大化区域 A 和区域 B 之间的方差。两个区域之间的方差过小,则提取出的图像会包含过多的身体深度信息,导致提取失败。因此,最大的方差意味着错误分类的概率最小。最佳分割阈值应使得区域 A 和区域 B 的方差最大化。最佳阈值 T 应满足以下等式:

$$T = \max_{0 \leq i \leq L-1} \{\sigma^2(t)\} \quad (7)$$

4 手部关节位置推断

要通过深度图像推断出正确的手部关节位置,首先需要提高腕部关节的正确识别率。正向运动学原理利用机器人运动学方程,以从该关节参数指定的值计算所述端部执行器的空间位置。以此为基础,本文提出了一种推断腕部关节的方法,即在测试者挥动手臂的过程中,通过肩部和肘部的弯曲和伸展角度,判断出腕部关节的位置。在确定腕部关节的前提下,可以精确地分割出手部的深度图像,利用质心公式求得手部关节坐标的空间位置。

Kinect 的真实世界坐标系如图 5 所示,该坐标信息表明了人体与坐标原点的距离 (x, y, z) 。

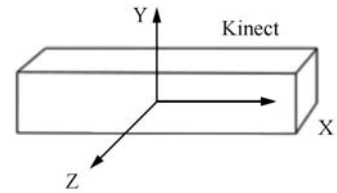


图 5 Kinect 3D 坐标系

要确定出肩部、肘部和腕部关节的几何关系(弯曲、伸展)应满足以下条件:

- 1) 测试者应当正面朝向 Kinect 并且沿着坐标系的 Z 轴方向移动,确保准确检测到人体;
- 2) 肩部和肘部关节的角度信息是确定的,即没有受到其他物体遮挡;
- 3) 手臂的运动方向应与坐标系的 YZ 平面平行,以确保腕部关节在 X 轴上的值与肘部关节的值相同。

关节间的角度与空间关系如图 6 所示,腕部空间坐标的计算公式如下:

$$W(z) = E(z) + \sin(\beta - \alpha)L \quad (8)$$

$$W(y) = E(y) - \cos(\beta - \alpha)L \quad (9)$$

$$W(x) = E(x) \quad (10)$$

式中: $W(x)$ 、 $W(y)$ 和 $W(z)$ 表示腕部的 x 、 y 和 z 值; $E(x)$ 、 $E(y)$ 和 $E(z)$ 表示肘部的 x 、 y 和 z 值; L 为前臂的长度;角度 α 是上臂弯曲/伸展的角度;角度 β 是前臂弯曲/伸展的角度。基于深度图像可以得到前臂长度,以及肩部和肘部的位置信息,因此可以计算出角度 α 的值,所以对于当前帧,只需要确定角度 β 的值就可以推断出腕部关节的位置。手臂的挥动可以看作是匀速平滑的运动,在肩部、肘部、腕部关节都处于追踪状态时,可以使用这些确定帧构建出手臂挥动模式的动态列表,对于需要推断出未知腕部关节位置的帧,可以将肘部的弯曲/伸展角度与列表进行匹配,获得角度 β 的值,然后计算出腕部关节的位置,同时更新列表。

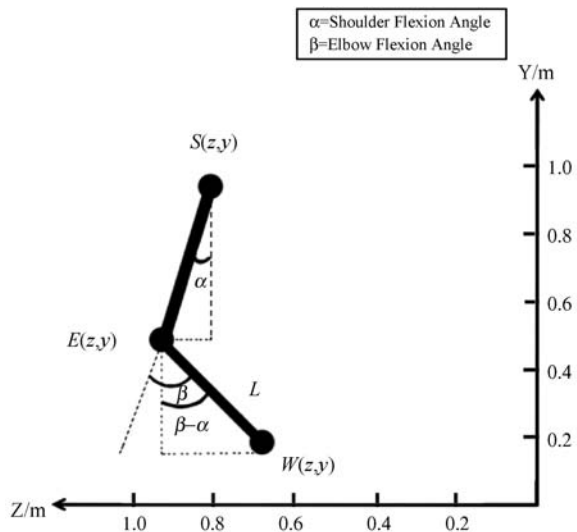


图 6 人体手臂关节在 YZ 平面的关系示意图

确定肘关节和腕关节的位置后,将手臂的深度图像进行二次分割,仅保留手部的深度图像。将图像转换为二值图像,使用质心公式求得手部关节的坐标。设二值图像为 $I(x,y)$,目标部分为 A ,目标的质心 (x_0,y_0) 定义如下:

$$I(x,y) = \begin{cases} 1 & (x,y) \in A \\ 0 & (x,y) \notin A \end{cases} \quad (11)$$

$$x_0 = \frac{\sum_{(x,y) \in A} xI(x,y)}{\sum_{(x,y) \in A} I(x,y)} \quad y_0 = \frac{\sum_{(x,y) \in A} yI(x,y)}{\sum_{(x,y) \in A} I(x,y)} \quad (12)$$

5 实验结果及分析

实验环境为:动作捕捉设备 Kinect for Windows 2.0,开发环境为 Visual Studio 2015 的 C#平台,Kinect SDK 和 Kinect Studio。

实验电脑配置如下:CPU 为 Intel Core i5 3.30 GHz,64 位处理器,4 GB RAM,视频采集卡 DX11,显卡为 NAIDIA Ge Force GT 940 M,内置 USB3.0 总线,OS 为 Windows 10。

本文选择了 10 名身高在 1.70 ~ 1.85 m 之间的参与者进行测试。Kinect 的放置高度为 0.8 m,Kinect 与测试者的距离为 4 m,测试者在 Kinect 的 Z 轴方向上前进并且匀速挥动手臂。在初始状态时保证测试者的肩部、肘部和腕部关节被正常追踪,从而建立第一次运动周期的手臂挥动动态列表。分析关节在“追踪”状态下和“推断”状态下 Kinect 读取出的 Z 值与本文提出方法得出的 Z 值之间的绝对误差,最终得到均值和标准差。在“追踪”状态和“推断”状态时 Kinect 和本文方法的部分输出如表 1 所示。

表 1 关节推断方法与 Kinect 返回读数的输出对比表 m

Z (Kinect 读数)	Z (本文方法读数)	绝对误差	检测的关节状态
3.778 530	3.777 999	0.000 531	追踪
3.777 813	3.771 467	0.006 346	追踪
3.776 693	3.765 336	0.011 357	追踪
3.721 065	3.701 811	0.019 254	追踪
3.738 104	3.697 985	0.040 119	追踪
3.738 098	3.678 147	0.059 951	推断
3.736 980	3.651 019	0.085 961	推断
3.734 640	3.615 120	0.119 520	推断
3.723 321	3.556 001	0.167 320	推断
3.707 803	3.539 961	0.167 842	推断

表 2 为关节推断方法 Y 值和 Z 值绝对误差的均值和标准差。可以看出,推断算法有一个较小的误差幅值。在关节处于“追踪”状态时,Y 轴和 Z 轴上的绝对误差分别为 0.0199 ± 0.0137 和 0.1270 ± 0.0875 。当 Kinect 处于“推断”状态时的绝对误差幅度比处于“追踪”状态时的绝对误差幅度更大,Y 值为 0.0686 ± 0.0186 ,Z 值为 0.2700 ± 0.0520 。在“推断”状态时,Kinect 无法准确检测到手部关节,只能通过深度图像推断其他关节来确定手部关节。深度图像的精确度越高,误差越小。本文将手部关节输出的坐标点位于手部深度图像中心 ± 0.02 m 范围内认为是一次正确识别,Kinect 与本文方法在“追踪”状态和“推断”状态时的准确率如表 3 所示。

表 2 关节推断方法绝对误差的均值和标准差

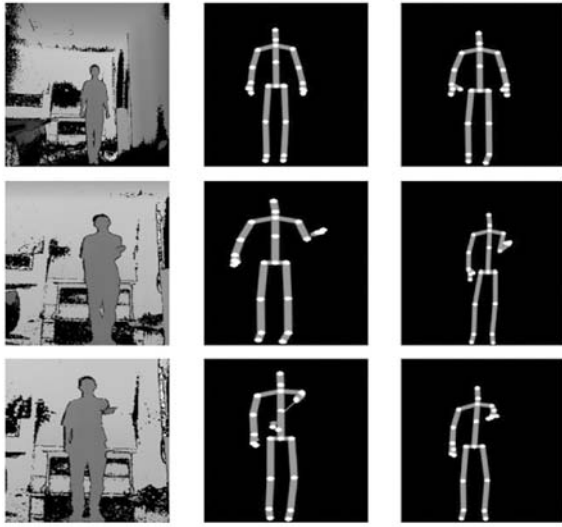
坐标	状态	均值	标准差
Y	追踪	0.019 972	0.013 735
	推断	0.068 673	0.018 651
Z	追踪	0.127 192	0.087 531
	推断	0.270 013	0.052 107

表 3 Kinect 与本文方法的准确率 %

方法	追踪状态准确率	推断状态准确率
Kinect	91.23	58.06
本文方法	92.21	76.89

本文方法与 Kinect 的骨骼追踪方法比较结果如图 7 所示。Kinect 的骨骼跟踪方法在人体轮廓清晰的情况下可以获得较准确的骨骼提取结果,如图 7(c)中的第 1 幅图像。在其他情况下,Kinect 对关节部位评估的准确度会有较大的误差,如图 7(b)的第 2 和第 3 幅图。本文方法的结果如图 7(c)所示,与 Kinect 相比,

本文方法可以在身体处于运动状态或有部分遮挡时获得更精确的关节定位。



(a) 原始深度图像 (b) Kinect 捕捉结果 (c) 本文结果

图7 人体关节捕捉结果对比

6 结 语

本文提出的手部关节位置推断方法能有效解决 Kinect 在无法正确追踪手部关节时识别精度较低的问题。本文方法基于深度图像信息进行关节的识别判断,避免了多余的数据采集,能够在保证实时性的前提下保持系统的运行效率;基于正向运动学公式,能有效判断出腕部关节的位置,提高了对腕部和手部关节位置的识别精度。下一步的工作是在大范围挥动手臂的情况下(肘部和腕部的 X 值不同)进行精确地关节捕捉,从而使 Kinect 更好地运用到体育分析和游戏娱乐等领域。

参 考 文 献

[1] Han J, Shao L, Xu D, et al. Enhanced computer vision with Microsoft Kinect sensor: A review[J]. IEEE Transactions on Cybernetics, 2013, 43(5): 1318 - 1334.

[2] Kohli P, Shotton J. Key developments in human pose estimation for Kinect[M]//Fossati A, Gall J, Grabner H, et al. Consumer Depth Cameras for Computer Vision. London: Springer, 2013: 63 - 70.

[3] Lin J C, Sun Q, Li G L, et al. Snap Blocks: a snapping interface for assembling toys blocks with XBOX Kinect[J]. Multimedia Tools and Applications, 2014, 73(3): 2009 - 2032.

[4] Xu X, McGorry R W, Chou L S, et al. Accuracy of the Microsoft KinectTM for measuring gait parameters during treadmill walking[J]. Gait and Posture, 2015, 42(2): 145 - 151.

[5] Wang Q F, Kurillo G, Ofli F, et al. Evaluation of pose tracking accuracy in the first and second generations of Microsoft Kinect [C]//2015 International Conference on Healthcare Informatics, 2015: 380 - 389.

[6] Osman S E K, Hembroff G, Matos M A. A novel Kinect-based algorithm for inferring the position of the lower body joints using human gait pattern[C]//2018 IEEE International Conference on Healthcare Informatics, 2018: 307 - 312.

[7] Srisen P, Auephanwiriyakul S, Theera-Umpon N, et al. Kinect joints correction using optical flow for weightlifting videos[C]//2015 Seventh International Conference on Computational Intelligence, Modelling and Simulation (CIMSIM), 2015: 37 - 42.

[8] Rosario H D, Belda-Lois J M, Fos F, et al. Correction of joint angles from Kinect for balance exercising and assessment[J]. Journal of Applied Biomechanics, 2014, 30(2): 294 - 299.

[9] 李昕迪,王云龙,何艳,等. 基于 Kinect 的人体单关节修复算法研究[J]. 自动化技术与应用, 2016, 35(4): 96 - 98, 120.

[10] Moen T S. Evaluation of a markerless motion capture system as a tool for sports movement analysis[D]. Oslo: University of Oslo, 2014.

[11] Wang C, Liu Z, Chan S C. Superpixel-based hand gesture recognition with Kinect depth camera[J]. IEEE Transactions on Multimedia, 2014, 17(1): 29 - 39.

[12] Pfister A, West A M, Bronner S, et al. Comparative abilities of Microsoft Kinect and Vicon 3D motion capture for gait analysis[J]. Journal of Medical Engineering and Technology, 2014, 38(5): 274 - 280.

[13] Trinh L A, Thang N D, Tran H H, et al. Human extraction from a sequence of depth images using segmentation and foreground detection [C]//Symposium on Information and Communication Technology, 2014: 178 - 185.

(上接第 76 页)

[7] Wei Q S, Gong B Z, Pathak S, et al. WAFTL: A workload adaptive flash translation layer with data partition[C]//Proceedings of the 2011 IEEE 27th Symposium on Mass Storage Systems and Technologies (MSST). IEEE, 2011: 1 - 12.

[8] Xu Z Y, Li R X, Xu C Z. CAST: A page-level FTL with compact address mapping and parallel data blocks [C]//2012 IEEE 31st International Performance Computing and Communications Conference (IPCCC). IEEE, 2012: 142 - 151.

[9] Kim Y, Brendan T, Gupta A, et al. FlashSIM: A simulator for NAND flash-based solid-state drives[C]//Proceedings of the 2009 1st International Conference on Advances in System Simulation. IEEE, 2009: 125 - 131.